RADIO CORPORATION OF AMERICA

Electronic Data Processing

CAMDEN, N. J.

RCA

NETWORK SCHEDULING CONCEPTS FOR PERT APPLICATIONS

Prepared by . . .

MANAGEMENT SCIENCE

NETWORK SCHEDULING CONCEPTS

FOR

PERT APPLICATIONS

Printed in U.S.A.
AT662

TABLE OF CONTENTS

NETWORK SCHEDULING CONCEPTS

FOR

PERT APPLICATIONS

SECTION I - INTRODUCTION

This booklet guides the reader, step by step, through the details of PERT. Special attention is given to the use of the basic building blocks of the network, network construction, and the PERT Algorithm.

The basic definitions and concepts of PERT along with a detailed description of computer requirements were provided in manual form as part of the PERT package released by Management Science.

The material in this applications aid, to be studied and/or presented to advantage, must be used in conjunction with the RCA PERT Booklet.

Planning and Scheduling - Past and Present

PERT is a management tool which is one of the first manifestations of the computer era in scientific management. In the days prior to the computer, the most widely used of management planning and scheduling tools was the Gantt Chart. The need for a more comprehensive planning tool was keenly felt by all who were engaged in the management of large-scale projects. It was not until the computer came into wide use that such a tool was feasible. The tool, PERT, replaced the bar graph of the Gantt Chart with a network. This network or arrow diagram depicts graphically much more information than a Gantt Chart.

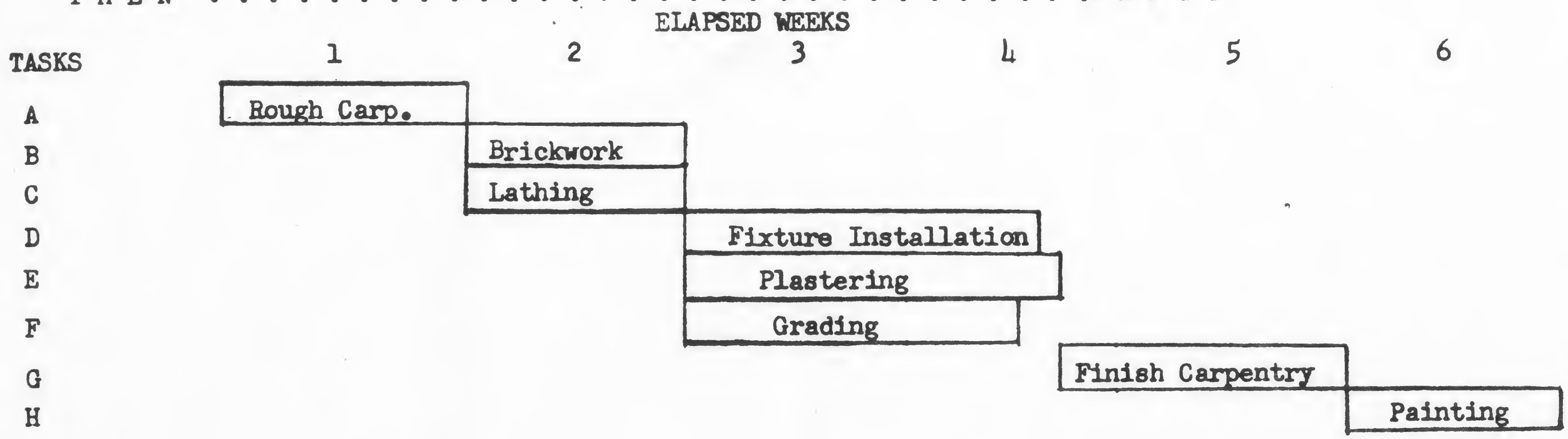
Figure 1 is a detailed PERT network. At this point, all that is intended is that the reader grasp how the Gantt Chart tasks are depicted in a PERT arrow diagram. The construction of the network will be discussed in detail when the various notations and rules for building a network have been covered.

Planning and Scheduling - PERT vs. Gantt

There are two important basic pieces of information which management must have to successfully plan a project. It must distinguish between planning and scheduling. Planning is the sequence of activities which must be performed to complete a project along with all the interrelationships between activities. Scheduling is the fitting of a plan to a time track or calendar. With the Gantt Chart (see Figure 2), planning and scheduling are presented simultaneously and are inseparable. Interrelationships are difficult to depict and the effect of changing time relationships is not economically feasible to assess, and the final results fall short of practical requirements.

With PERT these difficulties are overcome by attacking the problem both graphically and analytically. This is the important feature that makes PERT the powerful tool that it is. It enables the user to do planning and scheduling independently (see Figure 3). Slavish adherence to a prescribed scale (1 day = 1 block, etc.) that is necessary with a Gantt Chart is eliminated. With PERT the RCA 501 Computer performs the tedious calculations which convert time estimates into calendar dates and establish probabilities of meeting schedule dates. The resulting output is in quantitative terms and directs its main emphasis on job areas which are in most serious need of attention. With a PERT schedule, changes can be readily evaluated as to their overall effect on the total job.

T H E N . . . . G A N T T



N O W . . . . N E T W O R K

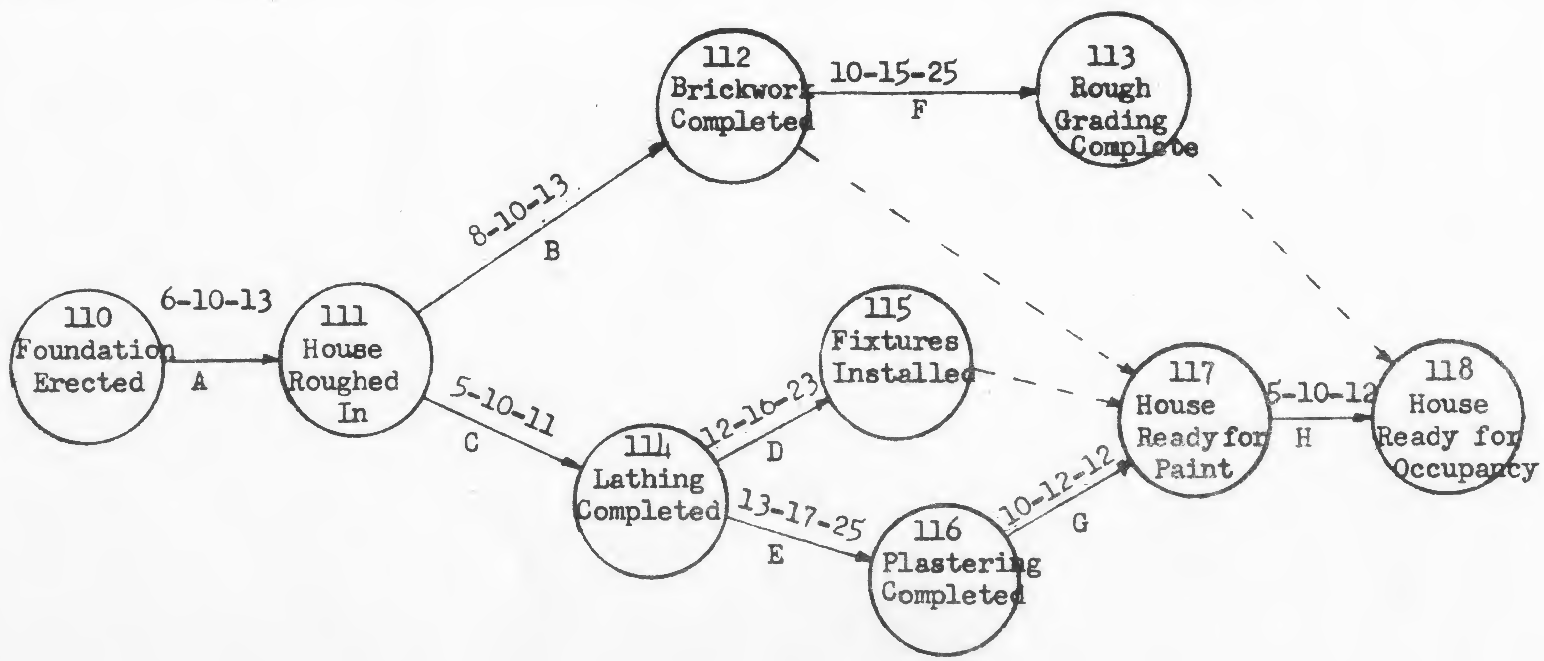


Figure 1

PLANNING

AND

SCHEDULING

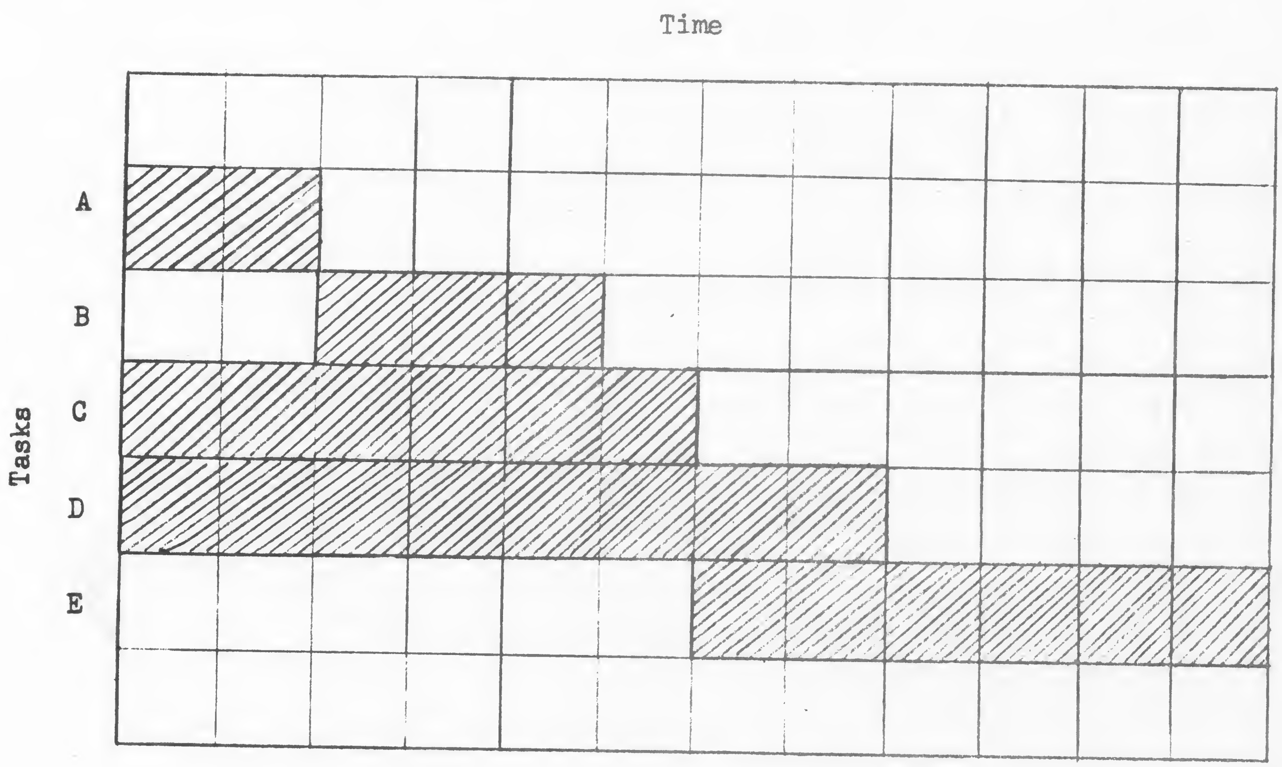
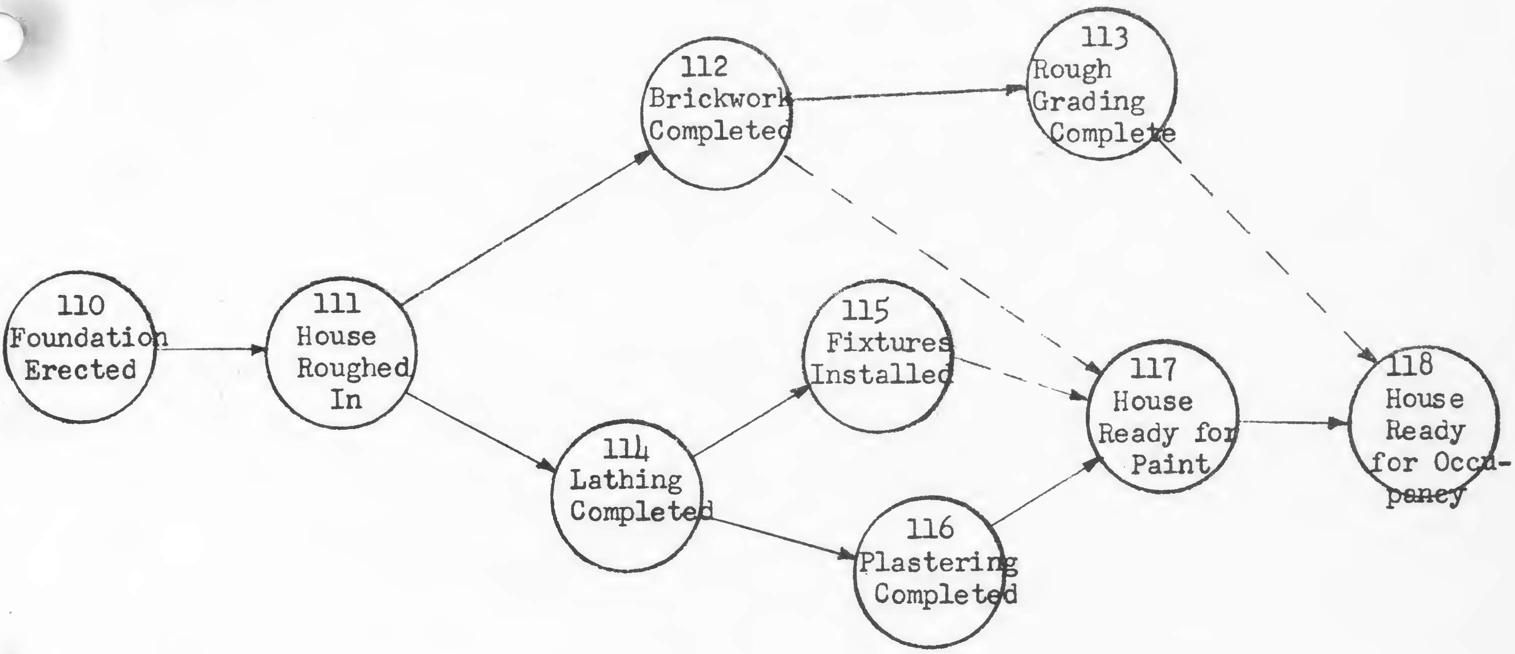


Figure 2

PLANNING



THEN

SCHEDULING

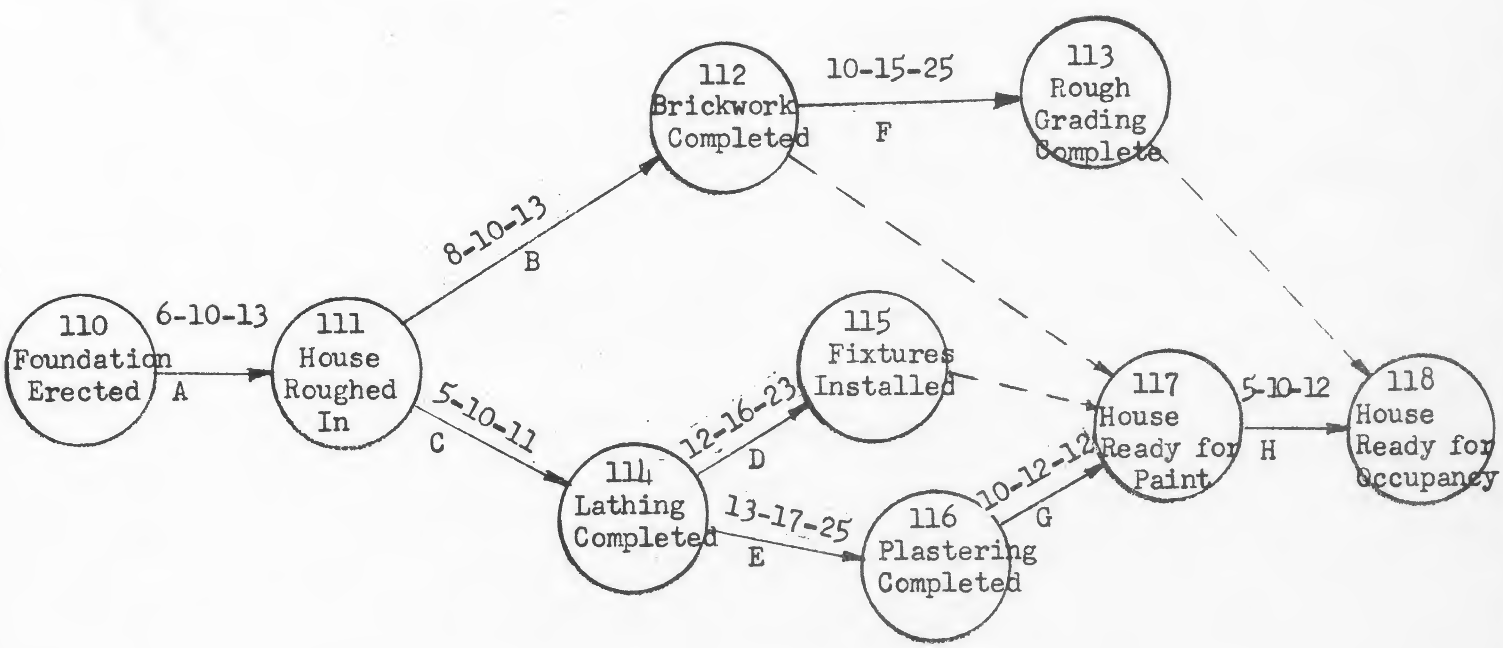


Figure 3

F E A T U R E S

1. Separates Planning from Scheduling

2. Pinpoints Relationships

3. Handles Doubts in Estimating

Figure 4

SECTION II - NETWORK BUILDING BLOCKS

Once the PERTing of a project has been decided upon, the most important phase of the ensuing work will be the construction of a precise arrow-diagram which is a faithful representation of all the events and inter-relationships comprising the entire project.

Basic Building Blocks of a PERT Network

The two building blocks of the PERT network are circles and arrows. Circles represent events, arrows represent activities and the times associated with their duration.

Figure 5 illustrates the relationship between arrows and circles. Graphically, it depicts the fact that the event "house ready for occupancy" takes place after the event "house ready for painting". The arrow represents the activity which occurs between events.

Network Time Estimates - Establishing Probability

The three numbers above the arrows are the activity time estimates or the estimated time between events. They are from left to right the optimistic, realistic and pessimistic estimates for the completion of the activity. These estimates enable PERT to provide planning and scheduling which is probabilistic in nature. Referring again to Figure 5, the planner's optimistic time estimate was $\frac{1}{2}$ week (the input is in tenths of weeks, 10 tenths = 1 week), the realistic 1 week, the pessimistic 1.2 weeks. Figure 6 illustrates how these estimates are used to establish probabilities.

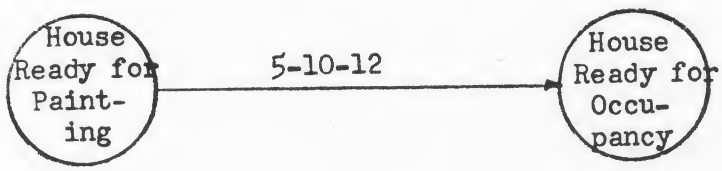
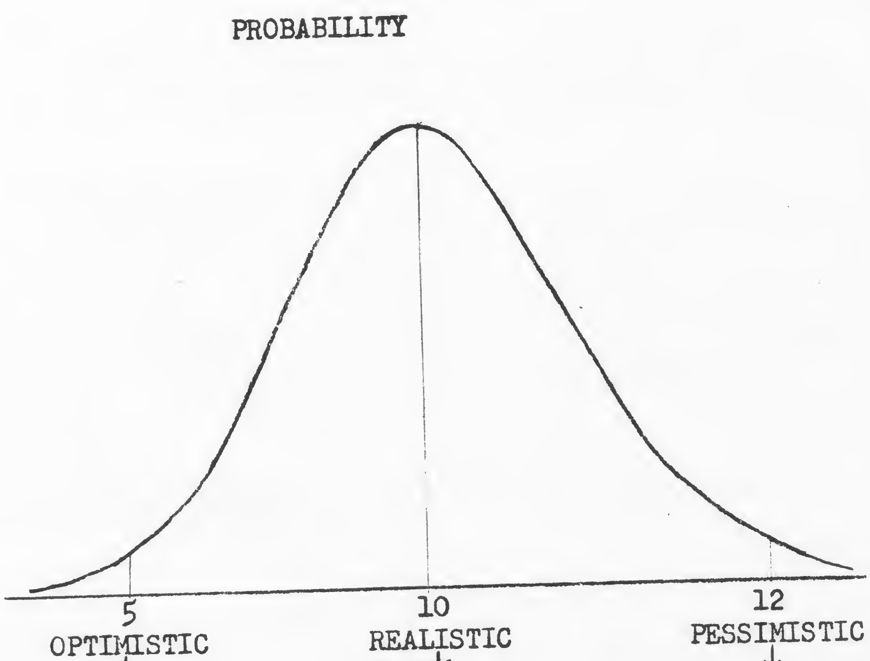


Figure 5



$$\text{EXPECTED TIME} = \frac{5 + (4 \times 10) + 12}{6} = \frac{57}{6}$$

$$\text{STANDARD DEVIATION} = \frac{12 - 5}{6} = \frac{7}{6}$$

Figure 6

SECTION III - REPRESENTING THE JOB

When constructing the flow diagram, it is important that certain relationships between the activities and events be maintained so that the resulting network can be converted into input data which the computer can accept.

Parallel Activities - The Dummy Activity

Consider the diagram in Figure 7. Activity B would be install plumbing (the B and other letters used are not requirements of the arrow diagrams, they are used here only to aid understanding). Activity A would be install lath. When these two activities are completed, the "house is completely lathed". Both activities, however, begin and end at the same mode or event. When presented as input to the computer, they are indistinguishable. This dilemma is circumvented by making use of the "dummy activity". The dummy requires no time to perform and hence does not enter into the calculation of the critical path. Figure 8 shows how the network appears when the dummy is added. The extra event and dummy activity enables the RCA 501 to distinguish between activity A and B and properly assess the contribution of each activity to the critical path. Dummies are usually represented as dashed lines. The activities A and B are said to occur in parallel.

Multiple Dependency

Figure 9 illustrates another situation in which the dummy can be put to good use. Referring to Figure 9 "C" depends upon the completion of "A" and "B" but "D" does not depend upon the completion of "A". Figure 10 shows how a dummy is used to diagram this condition.

The Constraint Situation

Often an activity will depend upon a restraint such as a delivery date. Figure 11 and 12 show how the dummy is used in this situation. It will be noted in Figure 12 that the use of the dummy will prevent "B" from starting until both "A" and "E" have been completed.

The 'OR' Node - The Short Path Indicator

The examples discussed thus far are all "and" nodes, i.e., before an activity can start from an event, all activities entering the node must be completed. Often in practice a job can proceed as soon as either job has reached the node. Consider Figure 13. We would like to represent the situation so "D" can start as soon as "C" and either "A" or "B" is completed. Figure 14 provides the solution. The cross hatching on "A" and "B" indicate that they have short path indicator codes. The shortest rather than the longest path will be chosen by the program (see PERT manual).

The above examples illustrate the use of the dummy activity and point out the essential role played by the dummy in constructing the network. All the dummies used thus far were zero time dummies.

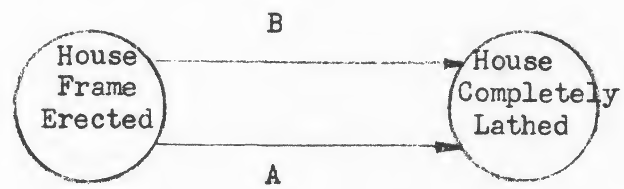


Figure 7

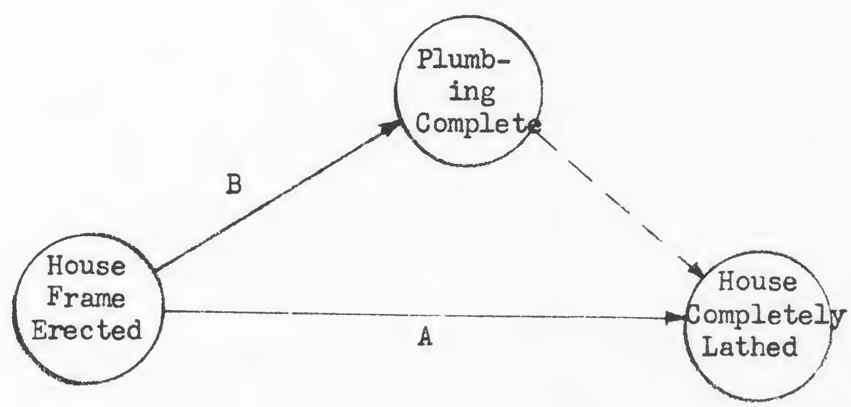


Figure 8

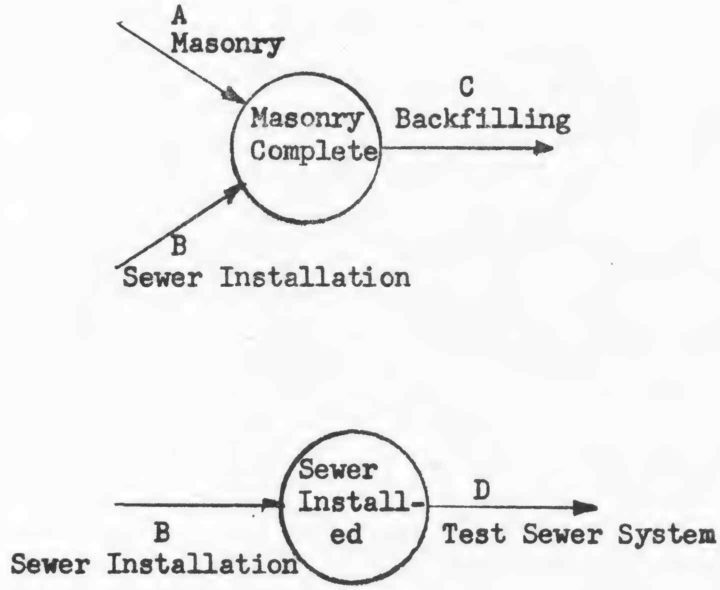


Figure 9

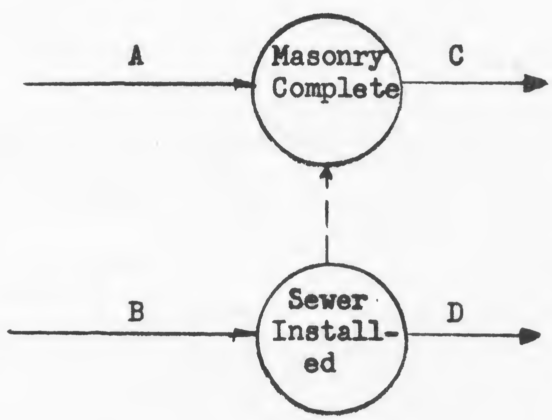
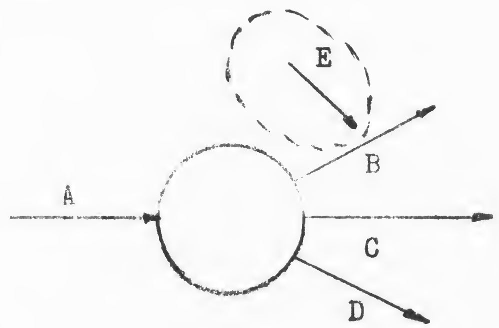


Figure 10



"B" has a delivery restraint "E"

Figure 11

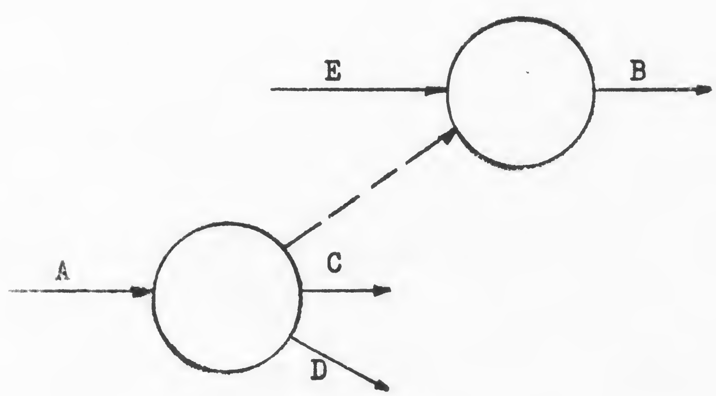


Figure 12

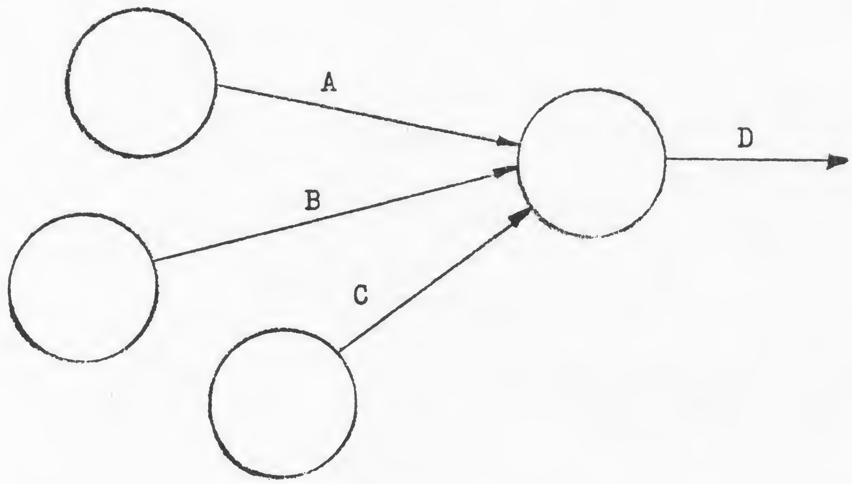


Figure 13

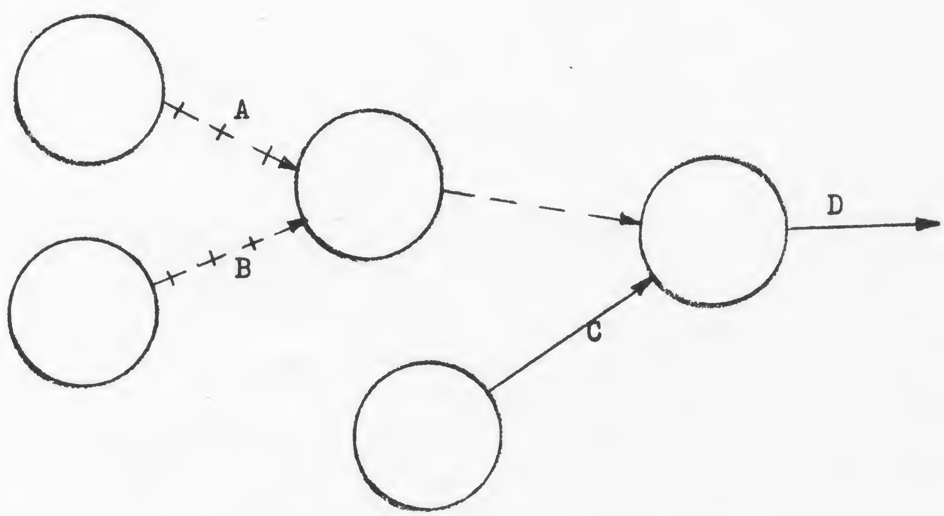


Figure 14

The Real Time Dummy-Lead Time

Instances arise, however, in which the dummy can assume an actual time value. This usually occurs when a lead time must be maintained for an event. Figure 15 illustrates the installation of a sidewalk. Installing forms cannot start as soon as digging begins but can start as soon as a certain lead time 'x' has elapsed. After digging is finished, the walk will not be finished until curing, pouring, etc., has taken place. Hence the lead time 'y'. 'x' and 'y' are dummies in that they do not represent any specific activity but a fixed, non-zero time is associated with them. In this sense they are real time dummies.

The examples described above are intended to aid the user in building a workable PERT network. Using these techniques individually or in combination the majority of PERT network problems can be solved.

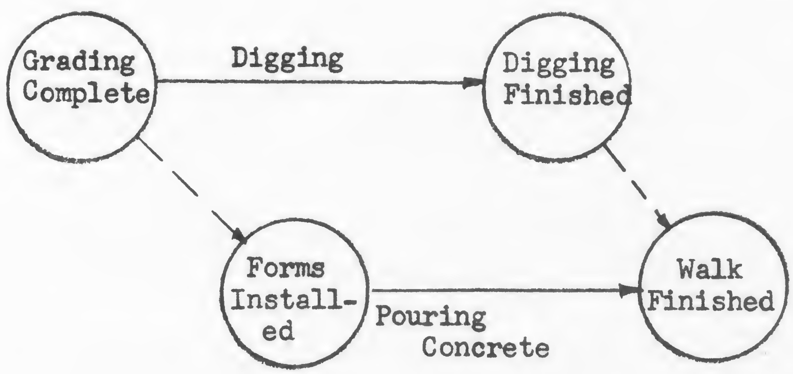

Grading Complete
Digging
Digging Finished
Forms Install-ed
Walk Finished
Pouring Concrete

Figure 15

SECTION IV - ACTUAL NETWORK CONSTRUCTION

A solution of the arrow diagram in Figure 1 will be used to illustrate the PERT algorithm. Before proceeding, however, it would be well to examine the general thought process involved in the original construction of the network.

How to Start - Sketch of Thought Flow

The project to be PERTed is the building of a house. A good rule to observe in constructing the network is to start at the terminal event. In our case, a logical terminal event would be "house is ready for occupancy". After fixing the terminal event, preceding events are considered. Before the house can be occupied, it must be painted and graded. These jobs would immediately precede the terminal event. Painting and grading are independent of each other and, therefore, can occur in parallel. Figure 16 illustrates this portion of the network. Use of the dummy imposes the condition that the house cannot be occupied until grading is complete. Before grading can begin, the brickwork must be completed. The brickwork can be done independently of events to this point. However, before the house can be painted, the brickwork must be finished. Hence we add a dummy between "brickwork completed" and "house ready for paint". Figure 17 is the network as it appears at this point.

In addition to the completion of the brickwork, the house will not be "ready to paint" until the fixtures are installed and the finish carpentry is done. The finish carpentry is an activity (activity G) and cannot begin until the event "plastering completed" has occurred. The installation of fixtures and the finish carpentry may be done simultaneously and are shown as parallel activities. These relationships are added to the existing network and appear in Figure 18.

The installation of fixtures and plastering both depend upon the lathing being completed. Since the activities involved occur in parallel and both depend upon the completion of a single event, they can be represented as in Figure 19. The remainder of the network is completed in a manner similar to the one just described. Figure 20 shows the network as it appears at the completion of the planning stage.

Scheduling the Completed Network

The scheduling stage involves the attaching of time estimates to the activities involved. At this time, the start date, final date, and any schedule dates are also affixed. Only one schedule date, July 10, 1961 which is the occupancy date, has been established for our network. Obviously caution should be exercised here for the final results can be no better than the time estimates supplied. The finished PERT diagram appears in Figure 1 (the letters are not required and are used here only to facilitate the explanation).

Once the arrow diagram, including time estimates for each activity, is completed, the input is then prepared as described in the PERT manual. The RCA 501 utilizing the PERT program executes the algorithm which solves the network.

An understanding of the discussion to this point is all that is required to effectively use PERT. The material that follows concerning the algorithm is added to satisfy the curiosity of those who wonder "how it really works" and to provide a more comprehensive background for using advanced techniques.

SECTION IV - ACTUAL NETWORK CONSTRUCTION

A solution of the arrow diagram in Figure 1 will be used to illustrate the PERT algorithm. Before proceeding, however, it would be well to examine the general thought process involved in the original construction of the network.

How to Start - Sketch of Thought Flow

The project to be PERTed is the building of a house. A good rule to observe in constructing the network is to start at the terminal event. In our case, a logical terminal event would be "house is ready for occupancy". After fixing the terminal event, preceding events are considered. Before the house can be occupied, it must be painted and graded. These jobs would immediately precede the terminal event. Painting and grading are independent of each other and, therefore, can occur in parallel. Figure 16 illustrates this portion of the network. Use of the dummy imposes the condition that the house cannot be occupied until grading is complete. Before grading can begin, the brickwork must be completed. The brickwork can be done independently of events to this point. However, before the house can be painted, the brickwork must be finished. Hence we add a dummy between "brickwork completed" and "house ready for paint". Figure 17 is the network as it appears at this point.

In addition to the completion of the brickwork, the house will not be "ready to paint" until the fixtures are installed and the finish carpentry is done. The finish carpentry is an activity (activity G) and cannot begin until the event "plastering completed" has occurred. The installation of fixtures and the finish carpentry may be done simultaneously and are shown as parallel activities. These relationships are added to the existing network and appear in Figure 18.

The installation of fixtures and plastering both depend upon the lathing being completed. Since the activities involved occur in parallel and both depend upon the completion of a single event, they can be represented as in Figure 19. The remainder of the network is completed in a manner similar to the one just described. Figure 20 shows the network as it appears at the completion of the planning stage.

Scheduling the Completed Network

The scheduling stage involves the attaching of time estimates to the activities involved. At this time, the start date, final date, and any schedule dates are also affixed. Only one schedule date, July 10, 1961 which is the occupancy date, has been established for our network. Obviously caution should be exercised here for the final results can be no better than the time estimates supplied. The finished PERT diagram appears in Figure 1 (the letters are not required and are used here only to facilitate the explanation).

Once the arrow diagram, including time estimates for each activity, is completed, the input is then prepared as described in the PERT manual. The RCA 501 utilizing the PERT program executes the algorithm which solves the network.

An understanding of the discussion to this point is all that is required to effectively use PERT. The material that follows concerning the algorithm is added to satisfy the curiosity of those who wonder "how it really works" and to provide a more comprehensive background for using advanced techniques.

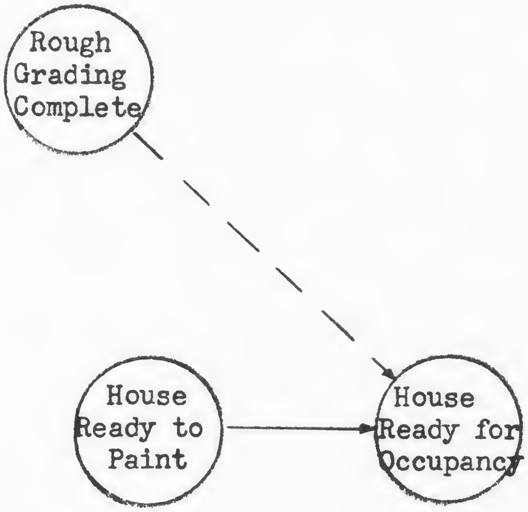


Figure 16

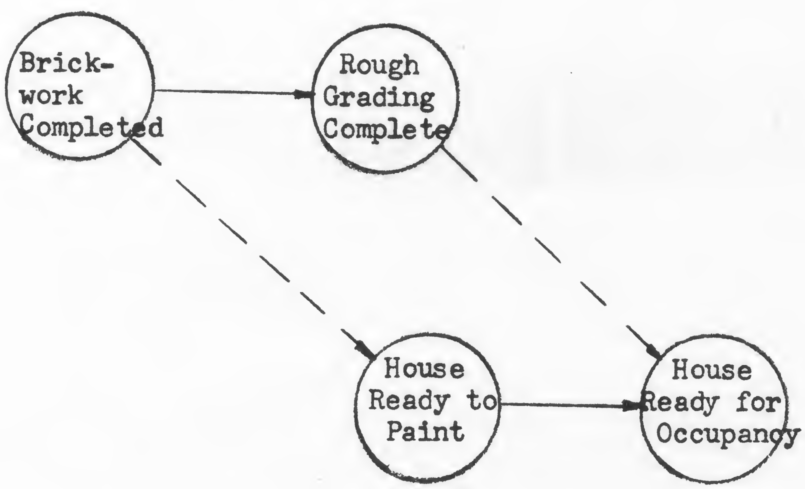


Figure 17

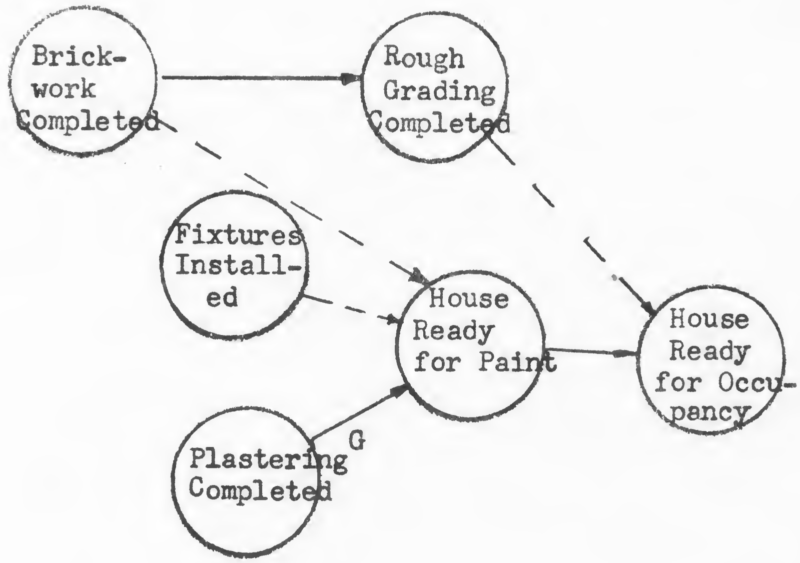


Figure 18

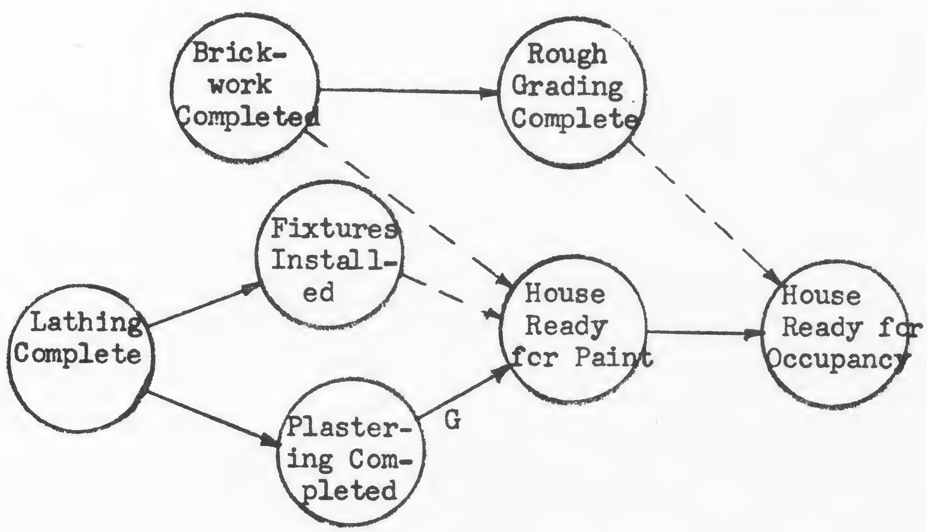


Figure 19

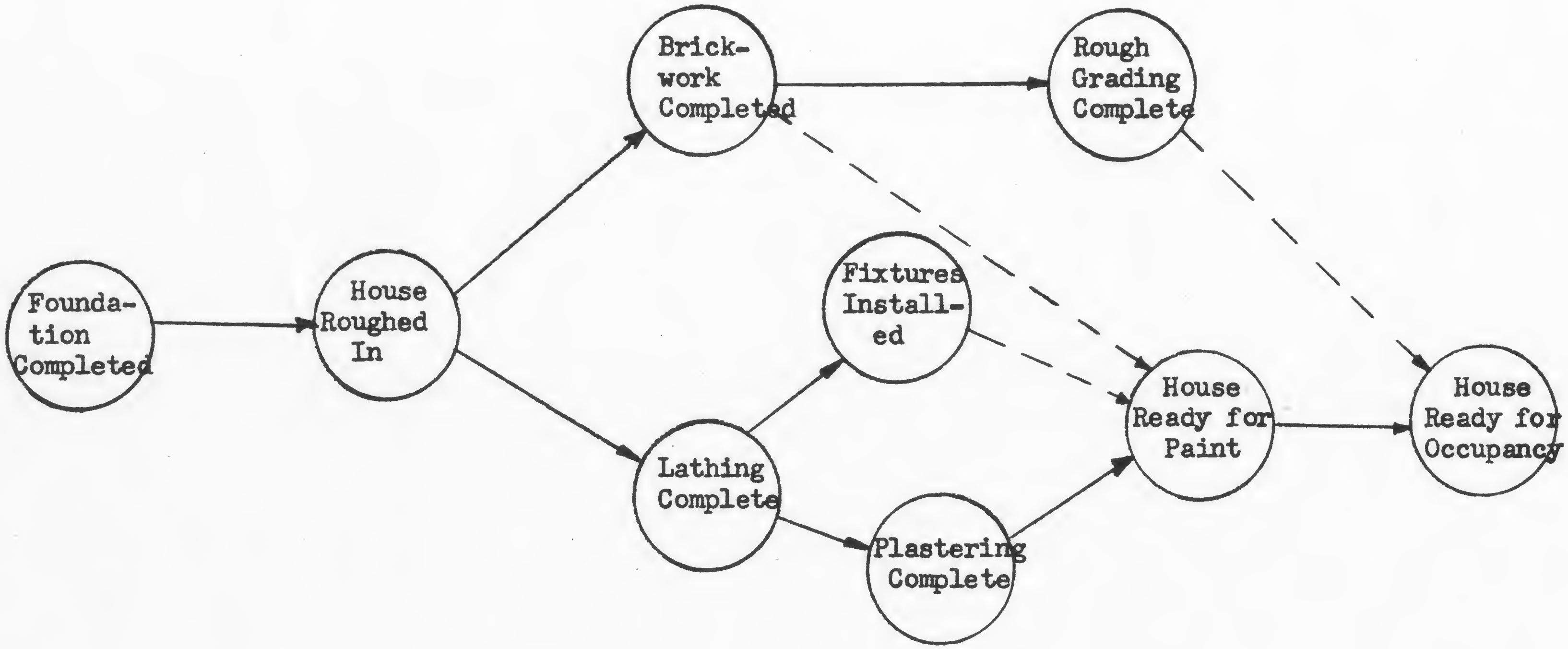

Founda-
tion
Completed
House
Roughed
In
Brick-
work
Completed
Rough
Grading
Complete
Fixtures
Install-
ed
Lathing
Complete
Plastering
Complete
House
Ready for
Paint
House
Ready for
Occupancy

Figure 20

V. A BRIEF DESCRIPTION OF THE PERT ALGORITHM

The preparation of the input for the problem is detailed in the PERT manual. The numbering of the events is arbitrary and need not be in any particular order.

The Computational Algorithm

Event numbers, however, must be unique. Once the data has been read in to the computer, the computing algorithm is essentially as follows:

1. A table is formed indicating the number of predecessors and successors of each event. Table 1 shows how the table appears initially for our problem.

	EVENT								
	110	111	112	113	114	115	116	117	118
PRECEDES	1	2	2	1	2	1	1	1	0
SUCCEEDS	0	1	1	1	1	1	1	3	2

TABLE 1

This table indicates that event 110 precedes one event and succeeds none. Event 117 precedes one and succeeds three. This can be easily verified by consulting the completed network of Figure 1.

The computer will now search the stored table for the event which has a 'zero' in its succeeds row (on the first search there will be only one such event, namely the starting event). In our example, it finds event 110. The program now notes that event 110 precedes only one event. This event is searched out and t_e (the expected duration) V (variance) and Ed (expected date) are calculated for the activity involved. In addition, the variance from preceding activities is accumulated (see manual for calculations of t_e, V, and Ed). These calculations are then stored and appear as in table 2.

ACTIVITY	t_e	V	Ed	V(acc)
110-111	7*	1*	7*	1*

TABLE 2

* The input is in tenths of weeks. The computer converts to days and calculations are rounded to the nearest day.

The program now subtracts one from the succeeds row under 111 in Table 1. This is done to enable the program to keep track of finished paths. Table 1 now becomes Table 1A.

	110	111	112	113	114	115	116	117	118
PRECEDES	1	2	2	1	2	1	1	1	0
SUCCEEDS	0	$\boxed{0}$	1	1	1	1	1	3	2

TABLE 1A

The program will now locate the next event which has a zero in its succeeds row. The event is 111. In this case, 111 has a two for its precedes entry, hence the RCA 501 will search for the two events which are successors of 111. The associated activities are 111-112 and 111-114. Calculations are made and the results placed in Table 2. Information now stored appears in Table 2A.

ACTIVITY	t_e	V	Ed	V acc.
110-111	7	1	000007*	1
111-112	7	1	000014	2
111-114	7	1	000014	2

TABLE 2A

V acc. is obtained by adding the variance that had been accumulated at the preceding event to the newly calculated variance of the event under consideration. For example, at event 111 the variance was 1. The variance for 111-112 was 1. Hence the variance accumulated at 112 is 1+1 = 2. This cycle continues until all succeeds numbers are zero and each event has been checked. Table 2B is the table at the completion of all calculations. The activity ordering numbers are inserted for referencing purposes later in the program.

Using the information in Table 2B, another table is constructed (Table 3) which contains:

1. The events in sequence
2. The expected date
3. The variance associated with the event
4. The activity number from Table 2

which will enable the program to locate the critical neighbor of the event under consideration.

* The number of days from the start date.

Activity Ordering	ACTIVITY	t_e	V	Ed	Vacc.
1	Start - 110	0	0	0	0
2	110 - 111	7	1	000007	1
3	111 - 112	7	1	000014	2
4	111 - 114	7	1	000014	2
5	112 - 113	11	3	000025	5
6	112 - 117	0	0	000014	2
7	113 - 118	0	0	000025	5
8	114 - 115	12	2	000025	4
9	114 - 116	12	2	000025	4
10	115 - 117	0	0	000025	4
11	116 - 117	8	0	000033	4
12	117 - 118	7	1	000040	5

TABLE 2B

Activity Number	EVENT	EXPECTED DATE	VARIANCE
1	110	000000	0
2	111	000007	1
3	112	000014	2
5	113	000025	5
4	114	000014	2
8	115	000025	4
9	116	000025	4
11	117	000033	4
12	118	000040	5

TABLE 3

Table 3 is constructed as the algorithm proceeds. Each time a node is reached, the table is inspected for the presence of the node number. If it is not present, it is placed in the table along with the other data that accompanies it. If the node number is present, the expected dates are compared and the largest one is placed in the table. In the case of event 117, three choices were possible -- 112-117, 115-117, and 116-117. The expected date of 116-117 being the largest was placed in the table along with '11' the activity ordering number in Table 2B from which the critical neighbor of 117 can be established. In this case, event 116.

The program now has all the information required for print out, with the exception of the latest date, slack, and the probability, if any, associated with the various events (probabilities are calculated only for those events which have schedule dates).

Establishing the Latest Date

For the terminal event, the latest date will be the schedule date if it is present -- otherwise it will be the expected date.

Establishing Slack

The slack for the final event equals the latest date (for final event) minus the expected date. The result is then filed. Since the schedule date for project is present, the latest date for event 118 is July 5, 1961, and the slack is a -5 days. The critical neighbor for 118 is then determined. The slack developed at 118 is then passed back to its critical neighbor, which is 117. The critical neighbor of each event was previously established and recorded in Table 3. Now event 117 is examined and a latest date and slack established for it. The latest date = Ed + Slack. If a schedule date is present at an event, it would be established as the latest date only if it were less than the latest date previously calculated by the formula Ld = Ed+Slack. Each event is examined in this manner until a latest date and slack have been established for each activity. Table 4 represents the final compilation.

EVENT	LATEST DATE	SLACK
110	-000005	-5
111	000002	-5
112	000024	10
113	000035	10
114	000009	-5
115	000028	3
116	000020	-5
117	000028	-5
118	000035	-5

TABLE 4

Calculating Probability

The only calculation remaining to be made is the probability of completing an event on schedule. Probabilities are calculated only for events which have schedule dates specified. A standard normal deviate is calculated and a table lookup performed to find the probability.

The expression which represents the standard normal deviate is

$$\frac{\text{Schedule date - Expected date}}{\sqrt{\text{Variance of event involved}}}$$

for event 118 we have

$$\frac{5}{\sqrt{5}} = \sqrt{5} = 2.24$$

The probability of exceeding this is .01; the .01 being established from a table lookup of standard normal deviates.

PERT Output Sorts

All the information required for the PERT print-out has now been calculated. The various sorts obtainable and the form of the edited information is available in the PERT manual. The slack sort and the form of the edited information is illustrated in Figure 21.

PROJECT REPORT

EVENT	CRITICAL NEIGHBOR	LANDMARK	ACTUAL DATE	EXPECTED DATE	LATEST DATE	SCHEDULE DATE	SLACK *	STD DEV *	PROB
110		FOUNDATION ERECTED		6/1/60	5/29/60		-5	0	
111	110	HOUSE ROUGHED IN		6/7/60	6/2/60		-5	1	
114	111	LATHING COMPLETED		6/14/60	6/9/60		-5	1.3	
116	114	PLASTERING COMPLETED		6/25/60	6/20/60		-5	2	
117	116	HOUSE READY TO PAINT		7/3/60	6/28/60		-5	2	
118	117	HOUSE READY FOR OCCUPANCY		7/10/60	7/5/60	7/5/60	-5	2.2	.01
115	114	FIXTURES INSTALLED		6/25/60	6/28/60		3	2	
113	112	ROUGH GRADING COMPLETED		6/25/60	7/5/60		10	2.2	
112	111	BRICKWORK COMPLETED		6/14/60	6/24/60		10	1.3	

Figure 21

* NOTE: The SLACK and STD. DEV. columns have been calculated in days to allow a better understanding of the method explained in the text. The computer printout would give these values in tenths of weeks.